# SPECIFICATION

Electronic Version 1.2.8

Stylesheet Version 1.0

# [ *METHOD FOR CREATING TEMPLATES TO CONVERT DATA INTO TARGET MARKUP LANGUAGES* ]

## Cross Reference to Related Applications

This application claims the benefit of U.S.C. § 119(e) of U.S. provisional application Serial No. 60/314,514, filed on August 22, 2001 entitled "Method for Creating Templates to Convert Data into Target Markup Languages" (Attorney Docket No. VIGN1330), which is hereby incorporated by reference as if set forth herein in its entirety.

## Background of Invention

[0001]     *Field of the Invention*

[0002]     The invention relates generally to methods for communicating data and more particularly to methods for generating templates for the conversion of XML and other non-display-formatted data to formats which are suitable for presentation on specific devices or classes of devices which utilize corresponding mark-up languages.

[0003]     *Background of the Invention*

[0004]     The Internet has become a powerful tool for disseminating information. Although the Internet was originally used to communicate scientific and technical data, increased numbers of service providers, improved applications for retrieving data and user-friendly interfaces have made it a useful tool for millions upon

millions of users. It is commonly used by people of all ages for transmitting and receiving messages, for research, for shopping and for surfing (browsing) the Internet.

[0005]     The increased popularity of the Internet is due in part to the amount of information which is available, and in part to the convenience with which people can access this information. Generally speaking, the information is accessed with browsers of various types. Traditionally, browsers have been computer-based applications which provide user interfaces to the Internet. These browsers are configured to request information from servers which are also connected to the Internet. The browsers typically communicate with the servers using a protocol known as hypertext transfer protocol, or HTTP. The servers, in turn, provided their information to the browsers via text and data which is marked or tagged so that the browser can format it for display to the user. This formatting information is typically provided through the use of hypertext markup language (HTML) tags that indicate which pieces of information should be formatted in a given manner.

[0006]     The use of HTML to provide formatting information to the browsers allows the same information to be transmitted to different types of browsers which have different capabilities, yet still be presented to the user in a readable/viewable fashion, adapted to the capabilities of the respective browsers. As a result, information providers do not have to tailor the format of their information to a particular type of browser and thereby limit the dissemination of the information. Instead, they can provide information in a single format which is accessible by a much greater user population.

[0007]     The accessibility of Internet-based information has increased further with recent developments in browser technology. Browsers can be incorporated into portable, wireless devices so that users can access the Internet without being tied to a PC. For example, some cellular phones and personal digital assistants (PDAs) incorporate Web browsers. Information which is transmitted to these browsers typically incorporates wireless markup language (WML), rather than HTML tags.

[0008]     One of the difficulties in accommodating all of these different types of client

devices is that each of them may use a different markup language which is specifically adapted to the capabilities of the corresponding device. A mechanism must therefore be provided to enable the conversion of generic data to the appropriate markup language for each of the devices. Typically, this mechanism is provided through a set of templates, wherein each of the templates is configured to convert the generic data into a corresponding markup language.

[0009]     There are several drawbacks to the use of these templates. One of these drawbacks relates to the fact that each of the templates is configured to convert data corresponding to one particular service into a markup language corresponding to one particular client device. Whenever it is desired to enable the transfer of data to a new client device, or to transfer data corresponding to a new service to the existing client, it is necessary to generate a new set of templates which is configured to make the required content transformations. It should be noted that, to support a single new client device, a set of templates corresponding to each of the available services is necessary. Likewise, in order to support a single new data service, a set of templates corresponding to each supported client device is necessary. Because the creation of new templates (typically using JSP or ASP technology) is difficult, the need to create a new set of templates for each new client device or service presents a daunting task.

## Summary of Invention

[0010]     One or more of the problems outlined above may be solved by the various embodiments of the invention. Broadly speaking, the invention comprises systems and methods for enabling the automatic creation of templates which are then used to convert service data from a non-displayable generic format to alternate formats which are suited to be displayed on various client devices. In a preferred embodiment, a master template corresponding to a set of devices is generated. The master template contains information relating to these devices which enable it to generate individual templates corresponding to specific service data and specific devices (or device classes). When it is desired to provide data corresponding to a new service to the client devices, the data corresponding to the new service is examined to identify its structure, specifically name-value pairs. A user is

prompted to accept a default label (the name from the name-value pair) or provide a new label corresponding to each pair. The user may select all or part of the service data to be used with the individual templates that will be generated. Based upon the service data and the information provided by the user, the master template is used to generate a service-specific template for each specific device class, wherein each template is configured to convert the service data into markup language data suitable for display on a corresponding device. The method thereby enables the automatic generation of templates for the conversion of the service data to markup language files that can be displayed on different client devices.

[0011]    In one embodiment, a method comprises examining service data and using a master template to generate service-specific templates for each of a plurality of devices or device classes. A master template is constructed to define a style for presentation of data on the devices. Service data such as XML data is examined to identify name-value pairs (XML tags and the information enclosed by the tags). The user is queried for labels that will be associated with these pairs. The XML tag corresponding to each pair may be used as a default label if no alternative is provided by the user. Based upon this information and the information contained in the master template, a plurality of device templates, each of which is associated with a specific type or class of devices, are generated. Each device template is configured to receive the service data and convert this data into a markup language format which is suitable for display on the associated device type. The method thereby eliminates the need to manually construct an individual device template for each device type whenever it is desired to provide a new data service.

[0012]    Another embodiment of the invention comprises a software application. The software application is embodied in a computer-readable medium such as a floppy disk, CD-ROM, DVD-ROM, RAM, ROM, database schemas and the like. The computer readable medium contains instructions which are configured to cause a computer to execute a method which is generally as described above. It should be noted that the computer readable medium may comprise a RAM or other memory which forms part of a computer system. The computer system would thereby be enabled to perform a method in accordance with the present disclosure and is

believed to be within the scope of the appended claims.

[0013]    Numerous additional embodiments are also possible.

## Brief Description of Drawings

[0014]    Other objects and advantages of the invention may become apparent upon reading the following detailed description and upon reference to the accompanying drawings.

[0015]    FIGURE 1 is a diagram illustrating the conversion of generic data into a format which is suitable for display on a selected client device.

[0016]    FIGURE 2 is a diagram illustrating the conversion of generic data into a plurality of different formats, each of which is suitable for display on a different client device.

[0017]    FIGURE 3 is a diagram illustrating a system according to one embodiment of the invention which is configured to generate a master template for the conversion of service data to various presentation formats.

[0018]    FIGURE 4 is a flow diagram illustrating a method in accordance with one embodiment of the invention.

[0019]    FIGURE 5 is a more detailed flow diagram illustrating a method in accordance with another embodiment of the invention.

[0020]    While the invention is subject to various modifications and alternative forms, specific embodiments thereof are shown by way of example in the drawings and the accompanying detailed description. It should be understood, however, that the drawings and detailed description are not intended to limit the invention to the particular embodiment which is described. This disclosure is instead intended to cover all modifications, equivalents and alternatives falling within the scope of the present invention as defined by the appended claims.

## Detailed Description

[0021]    A preferred embodiment of the invention is described below. It should be

noted that this and any other embodiments described below are exemplary and are intended to be illustrative of the invention rather than limiting.

[0022]    Broadly speaking, the invention comprises systems and methods for facilitating the communication of data which must be converted into a markup language in order to make it suitable for display on a receiving device. The present techniques allow templates to be quickly created for the conversion of data such as XML-formatted data into markup languages which are suitable for use with one or more corresponding devices. For example, information which is formatted as XML data may need to be converted to HTML before it can be viewed on a full-size browser such as Netscape. If, on the other hand, the data is to be transmitted to a wireless device, the data may need to be converted to a particular form of WML in order to be properly displayed on that device.

[0023]    Instead of manually generating individual templates to convert generic data for each service into each markup language which may need to be provided to a client device, a master template can be used to generate an individual template for each group of similar devices requiring a specific markup language and relying on a specific user interface pattern. Each individual template corresponds to a particular device type and a particular service. Individual templates corresponding to new data services can therefore be automatically generated.

[0024]    Raw data may be delimited in a way that it is not easily human-readable. For example, a series of values may be separated by semicolons or some other delimiter (e.g., "first name; last name; ..."). When a database engine reads this data, it is aware based upon the database schema that the first value corresponds to a first variable, the second value corresponds to a second variable, and so on (e.g., the first value is a first name, the second value is a last name, ...).

[0025]    In XML, the values are delimited by text tags which identify the corresponding variables. The tags have the form "<tag_name>" and "</tag_name>", where the former delimits the beginning of the value and the latter delimits the end of the value. For example, if the variable "first_name" has the value "John", this would typically appear in XML as <first_name>John</first_name>".

[0026]     XML differs from earlier in data formats in that it is easily human-readable, and that it is easy to program an application that understands the XML data and does something with it. An application could, for example, read data formatted as described above and identify the first_name and last_name values, handling each one in a manner which is appropriate for the application. This may consist of re-ordering the names, presenting them in a particular font, or using or manipulating the values in any other way. The application simply has to look for the appropriate tags and read the information which is enclosed by them. There is no need for special database drivers or software which allow the data to be read.

[0027]     The use of tags in XML is very similar to the use of tags in HTML. In XML, however, the tags are used only for the purposes of delimiting and identifying data. The tags do not define presentation information, such as font, color, position on a page, etc. In other words, the tags do not serve a markup purpose as they do in HTML and other markup languages. Below is an example of XML data for a weather reporting service.

[0028]     <weather_report>

[0029]     <day_of_week>Monday</day_of_week>

[0030]     <temperature>75</temperature>

[0031]     <wind_speed>10</wind_speed>

[0032]     </weather_report>

[0033]     This data corresponds to a service which provides data relating to particular weather conditions. The data for a given report is preceded by the opening tag <weather_report> and is followed by the </weather_report> tag. The individual pieces of data within the report are each enclosed by tags identifying the data.

[0034]     It should be noted that, for the purposes of this disclosure, the term "unformatted data" will be used to refer to XML or similar data despite the use of identifying tags. (This data could alternately be referred to as "non-display-formatted" data.) This data is referred to as unformatted because it does not

contain information relating to the manner in which the data should be formatted for the purposes of presentation. For example, it does not have tags identifying tables, cells within tables, fonts, colors, or other information that would be used to control the presentation of the data to a user.

[0035]     Referring to FIGURE 1, a diagram illustrating the conversion of generic data into a format which is suitable for display on a selected client device is shown. As shown in this figure, generic, unformatted data corresponding to a selected service is retrieved from a storage device 11. This data may be one of several forms which are, for the purposes of this disclosure, considered to be unformatted. That is, they are not in a format which is suitable for display on the client device. The data may, for example, be in the form of a table (e.g., wherein individual cells are separated by commas, spaces or other delimiters) or, more commonly, in the form of XML data. The unformatted data is converted, using XSL template 12 into formatted data which is suitable for display on a client device 13. (XSL is also sometimes referred to as XSLT -- extensible stylesheet language transformation.)

[0036]     As indicated above, the server must be capable of providing data to a variety of different types of clients. For example, the client may be a PC running a full-scale Internet browser such as Netscape, or it may be a cellular phone which has a browser with limited screen area and features. The client device may even be a two-way pager which can only display one or two lines of text at a time. A separate XSL template is needed for each of these devices. This is illustrated in FIGURE 2. In this figure, data from storage device 21 may be processed by a first template 22 to produce a set of HTML-formatted web pages 23 that can be displayed on the browser running on PC 24. On the other hand, the data from device 21 may be processed by template 25 to generate WML data 26 which is suitable for display on the browser of cellular phone 27. Likewise, the data can be processed by template 28 to produce data 29 which can be displayed within the device limitations of two-way pager 30. (It should be noted that FIGURE 2 depicts only a few of the many different devices and corresponding markup language is for which the server may be needed to provide data.)

[0037]    One embodiment of the present intention involves the use of a master template from which individual templates can be generated. These individual templates can then be used to convert unformatted (e.g., XML) data into a format which is suitable for presentation via a corresponding client device. This is illustrated in FIGURE 3.

[0038]    FIGURE 3 is a diagram illustrating a system according to one embodiment of the invention which is configured to generate a master template for the conversion of service data to various presentation formats. In this embodiment, an application 41 is configured to receive unformatted data 42 corresponding to a data service, examine this data and, based upon user input and a master template 43, generate a individual, service/device-specific templates 22.

[0039]    It should be noted that, for the purposes of this disclosure, identical items in the figures may be indicated by identical reference numerals followed by a lowercase letter, e.g., 22a, 22b, and so on. The items may be collectively referred to herein simply by the reference numeral.

[0040]    Master template 43 contains information that defines the manner in which data for the selected service is presented, as well as information on the presentation capabilities of the client devices. Master template 43 is not intended to be used to directly convert unformatted data to a particular markup language. It is instead intended to provide building blocks from which the individual templates (which are intended to convert the unformatted data into formatted data) can be constructed. The master template is used in conjunction with a software application (a "wizard") which identifies the name-value pairs in the service data and selects the building blocks from the master template which are necessary to generate the formatting for these name-value pairs.

[0041]    "Name value pairs is used here to refer generally to the generic format of the service data. It should be noted that the data may be in a more sophisticated format in some instances. For example, wind data could be presented in XML as

[0042]    <WindSpeed>5</WindSpeed>

[0043] <WindDirection>NW</WindDirection>

[0044] or

[0045] <Wind>

[0046] <Speed>5</Speed><Direction>NW</Direction>

[0047] </Wind>

[0048] The template wizard can be configured to handle either format.

[0049] It should also be noted that the XML data may contain pictures or other multimedia content. For instance, the weather conditions "Sunny" and "Cloudy" could be represented by sunshine or cloud icons, respectively. The wizard can be configured to display the icons on devices which have the appropriate graphics capabilities, while the words (e.g., "Sunny" or "Cloudy" could be displayed on devices which don't have the capability to display icons.

[0050] The master template follows a predetermined style. When the master template is written, the author chooses the manner in which data will be displayed. For example, it may be written to display name–value pairs as rows in a table. Different types of data may be handled differently. For instance, report headings may be displayed in a larger font than other data. The style may be adapted to each of the devices for which the master template will be used to create individual templates since the formatting for each type of data is defined in a separate building block.

[0051] Referring to FIGURE 4, a flow diagram illustrating a method in accordance with one embodiment of the invention is shown. In this embodiment, the method essentially comprises examining data which corresponds to a particular type of service, assigning appropriate labels to the different pieces of data provided via the service, retrieving the corresponding building blocks from the master template and assembling the building blocks to generate templates which can be used to create markup language files for corresponding devices.

[0052] Referring to FIGURE 5, a more detailed flow diagram illustrating a method in

accordance with another embodiment of the invention is shown. In this embodiment, the service data is formatted as XML data. This data is examined to identify name-value pairs which are present. For each of these pairs (or each name associated with one of these pairs), the user the first determines whether or not it is desired to retain the name-value pair. It may not be desirable to provide all of the information to the client devices, so some of the data may simply be discarded. In this manner, the data may be filtered. If a particular name-value pair is to be retained, the user is presented with the name (the XML tag) from the XML file and is given the option of accepting this name or modifying it for purposes of the individual markup languages. For example, if the tag is "day", the user may choose to use a tag which is more informative, such as "day of the week". Once the user has selected either the default or a new label, that label is used for the corresponding name-value pair. Based on the type of the name-value pair, the appropriate building block for each device is selected from the master template. The building blocks for each device are then assembled into a template for that device. Each template is specific to the corresponding device and the particular service associated with the data. In the preferred embodiment, a services wizard application is configured to examine the data for a new service and to automatically generate service-specific templates for each of the device types for which building blocks are defined in the master template. The individual templates may correspond to individual devices, or to classes of devices which can utilize data that is formatted according to the same markup language and device presentation capabilities. In alternative embodiments, the individual templates may be generated as needed rather than automatically upon completion of the master template.

[0053]

In the preferred embodiment, there is a single master template. The master template defines a single style (which may vary somewhat from device to device) which is embodied in the individual, service/device-specific templates. When it is desired to support a new type of client device, the master template must be updated (or a ew one written) to include building blocks corresponding to that device. These building blocks can then be used to define a service-specific

template for the new device. It is contemplated that, in an alternate embodiment, several master templates may be provided, wherein each defines a different style for the presentation of service data. A user could then be prompted by the wizard application to select a style (master template) according to which the individual templates will be generated.

[0054]      The present invention may provide a number of advantages over prior art methods. For example, although XML provides a mechanism for enabling the definition, transmission, interpretation, etc. of data between disparate applications and thereby simplify the interfaces between such applications, XML is widely regarded as a very unforgiving language. Consequently, it is typically very tedious and time consuming to manually construct the templates necessary for the generation of markup-language data which is suitable for display on the different client devices to which a server may provide data. The time and cost associated with the generation of these templates can make XML-based server solutions unattractive to users in spite of other advantages they might provide. By providing a means for the automatic generation of templates to accommodate new data services, the present methods may eliminate this disadvantage.

[0055]      The benefits and advantages which may be provided by the present invention have been described above with regard to specific embodiments. These benefits and advantages, and any elements or limitations that may cause them to occur or to become more pronounced are not to be construed as a critical, required, or essential features of any or all of the claims. As used herein, the terms "comprises," "comprising," or any other variations thereof, are intended to be interpreted as non-exclusively including the elements or limitations which follow those terms. Accordingly, a process, method, article, or apparatus that comprises a list of elements does not include only those elements but may include other elements not expressly listed or inherent to the claimed process, method, article, or apparatus.

[0056]

While the present invention has been described with reference to particular embodiments, it should be understood that the embodiments are illustrative and

that the scope of the invention is not limited to these embodiments. Many variations, modifications, additions and improvements to the embodiments described above are possible. Particularly, these variations may include computers or other data processing devices, computer readable media (such as floppy disks, CD-ROMs, DVD-ROMs, etc.,) storage devices, computer memories and the like which contain software, firmware or other programming embodying the foregoing methods. It is contemplated that these variations, modifications, additions and improvements fall within the scope of the invention as detailed within the following claims.